



FACHHOCHSCHULE AACHEN, CAMPUS JÜLICH

FACHBEREICH 09 - MEDIZINTECHNIK UND TECHNOMATHEMATIK
STUDIENGANG ANGEWANDTE MATHEMATIK UND INFORMATIK

SEMINARARBEIT

Anforderungsanalyse und Konzeptionierung einer Webanwendung zur strukturierten Darstellung von Lernmaterialien

Autor:

Theresia Rupprecht, 3276311

Betreuer:

Prof. Dr. Voß

Marcel Nellesen, M.Sc.

Aachen, 15. Dezember 2022

Zusammenfassung

Es ist das Ziel in der vorliegenden Arbeit eine interaktive Webplattform zu konzeptionieren. Diese soll Lernmaterialien strukturiert darstellen und verschiedene Themenbereiche abdecken. Das Projekt soll Themen, wie das Forschungsdatenmanagement einfach und verständlich zugänglich machen. Je nach Hintergrundkenntnissen werden passende Lernmaterialien vorgeschlagen.

Zuerst wurde eine Anforderungsanalyse durchgeführt, deren Eigenschaften wurden auf Basis von Use Cases definiert und in funktionale und nicht funktionale Anforderungen unterteilt. Als nächster Schritt wurde der aktuelle Stand der Technik betrachtet. Dafür wurden die Systeme Moodle, Codecademy und OrcaNRW verglichen. Es wurde untersucht welche Anforderungen von den Systemen erfüllt werden und welche nicht. Der Vergleich zeigte, dass keine Plattform die zuvor definierten Anforderungen erfüllt. Anhand dieser Feststellung wurde eine Konzeption für eine neue Plattform erstellt. Hierbei wurden als Erstes die Attribute eines Materiales herausgearbeitet und erklärt. Als Zweites wurde ein Datenbankmodell mithilfe eines Entity-Relationship Modell erstellt. Drittens wurde ein erster Entwurf für die Benutzeroberfläche erstellt. Perspektivisch wird die Realisierung im Rahmen einer Bachelorarbeit dargestellt.

Eidesstattliche Erklärung

Hiermit versichere ich, dass ich die Seminararbeit mit dem Thema

Anforderungsanalyse und Konzeptionierung einer

Webanwendung zur strukturierten Darstellung von Lernmaterialien

selbstständig verfasst und keine anderen als die angegebenen Quellen und Hilfsmittel benutzt habe, alle Ausführungen, die anderen Schriften wörtlich oder sinngemäß entnommen wurden, kenntlich gemacht sind und die Arbeit in gleicher oder ähnlicher Fassung noch nicht Bestandteil einer Studien- oder Prüfungsleistung war.

Ich verpflichte mich, ein Exemplar der Seminararbeit fünf Jahre aufzubewahren und auf Verlangen dem Prüfungsamt des Fachbereiches Medizintechnik und Technomathematik auszuhändigen.

Name: Theresia Rupprecht, 3276311

Aachen, den 15. 12. 2022



Unterschrift der Studentin / des Studenten

Inhaltsverzeichnis

1	Einleitung	1
1.1	Motivation	1
1.2	Zielsetzung	1
2	Anforderungsanalyse	3
2.1	Use Cases	3
2.2	Funktionale Anforderungen	6
2.3	Nicht-funktionale Anforderungen	8
3	Stand der Technik	10
4	Konzeption	13
5	Fazit	21
A	Literaturverzeichnis	23
B	Tabellenverzeichnis	25
C	Abbildungsverzeichnis	26

1 Einleitung

1.1 Motivation

Wer sich zu einem bestimmten Thema informieren möchte muss im Internet nur eine Suchanfrage tätigen und bekommt eine unüberblickbare Anzahl an Suchtreffern. Oft stellt sich dann aber heraus, dass es sich nicht um wirklich passende Ergebnisse handelt. Möchte man sich beispielsweise über das Forschungsdatenmanagement informieren, so findet man zwar schnell eine Definition, was genau das Wort beschreibt, aber einen passenden Einstieg in das Thema zu finden, ist deutlich schwieriger. Betrachtet man noch die jeweiligen Vorkenntnisse und Hintergründe der suchenden Person, ist ein richtiger Treffer bei den ersten Suchergebnissen äußerst unwahrscheinlich. Aus diesem Grund werden in Arbeitsgruppen Listen, mit passenden Tutorials, Videos und Lernmaterialien, geführt. Diese werden dann, zum Beispiel, zur Einarbeitung von neuen Mitarbeitern genutzt, denen Vorkenntnisse fehlen. Das ist zwar eine funktionierende Methode, jedoch immer noch sehr umständlich. Aus diesem Grund beschäftigt sich diese Arbeit mit der Analyse der Anforderungen einer Webanwendung, die Lernmaterialien strukturiert darstellt. Auf Basis der unternommenen Analyse wird dann eine passende Webanwendung entworfen.

Die entstandene Webplattform soll im Kontext von National High Performance Computing Center for Computational Engineering Science (NHR4CES) eingesetzt werden. NHR4CES ist eine Kooperation zwischen der Technische Universität Darmstadt und der RWTH Aachen. Das Ziel dieser Zusammenarbeit ist es High Performance Computing (HPC) und Forschungsdatenmanagement zu kombinieren [1].

1.2 Zielsetzung

Das Ziel der Webanwendung ist, eine Sammlung an Lernmaterialien leicht zugänglich strukturiert bereitzustellen. Hierbei sollen die Inhalte übersichtlich dargestellt werden und verschiedenen Themengebieten zugeordnet sein. Außerdem soll es für verschiedene Nutzergruppen vorgegebene Archetypen geben, die Vorkenntnisse der Gruppen berücksichtigt und ein spezifisches Lernziel definiert. Um alle wichtigen Anforderungen zu berücksichtigen, werden als Erstes verschiedene Use-Cases betrachtet. Mithilfe dieser Ergebnisse werden anschließend die funktionalen und nicht-funktionalen Anforderungen aufgestellt. Danach

wird der aktuelle Stand der Technik betrachtet und die Unterschiede der neuen Plattform zu bereits Bestehenden erläutert.

Diese Arbeit dient als Grundlage für eine Implementierung, deswegen wird anschließend noch eine Konzeption erstellt. Auf Basis des Entity-Relationship Diagramm erfolgt die Erstellung des Datenbankmodells. Im Anschluss wird noch die Benutzeroberfläche der Plattform entworfen. Abschließend erfolgt ein Ausblick über die weitere Zukunft dieses Projektes.

2 Anforderungsanalyse

Dieses Kapitel beschäftigt sich mit der Anforderungsanalyse der Webplattform. Um alle wichtigen Szenarien abzudecken, werden mithilfe von Use Cases Anforderungen definiert. Diese werden anschließend in funktionale und nicht-funktionale Anforderungen unterteilt und erläutert.

2.1 Use Cases

Use Cases helfen dabei zu definieren, was für Aufgaben die Webplattform erfüllen soll. Bei einem Use-Case werden die Teilschritte einer Aufgabe nicht berücksichtigt, sondern es geht um eigenständige Arbeitsprozesse, die ein Ergebnis liefern. Die genauen Abläufe werden erst später betrachtet. Mithilfe eines Use Case Diagramms kann eine gute Übersicht über das Gesamtsystem gewonnen werden [2, S.63-64].

Abbildung 2.1 zeigt das Use Case Diagramm der Webplattform. Ein einzelner Use Case ist mit einer Ellipse dargestellt. Ein Use Case schließt alle damit verbundenen Aktionen mit ein. So ist zum Beispiel bei der Suche eines Materials die Suche in der Datenbank genauso mit eingeschlossen, wie das Aufrufen der Suchfunktion. Use Cases müssen von Akteuren ausgelöst werden. Akteure sind in diesem Fall der Nutzer und der Administrator. Durch den Pfeil vom Administrator zum Nutzer wird eine Generalisierung dargestellt. Das bedeutet, dass ein Administrator auch ein Nutzer ist, aber nicht umgekehrt. Als System wird in unserem Fall die Webplattform verstanden, da diese die Use Cases realisieren soll [3, S.242-251].

Der nachfolgende Abschnitt erläutert die vier Use Cases und beschreibt deren Abfolge.

Nutzer möchte sich über ein bestimmtes Themengebiet informieren

Der erste Anwendungsfall tritt ein, wenn ein Nutzer sich über ein bestimmtes Thema informieren möchte. Dafür können die verschiedenen Archetypen benutzt werden. Ein Archetyp beschreibt die Personengruppen, die die Plattform benutzen werden. Außerdem hat jeder Archetyp ein Thema, über das er sich informieren möchte. Wenn sich ein Nutzer beispielsweise über das Thema Künstliche Intelligenz informieren möchte, dann kann er bei der Übersicht der Archetypen schauen, welcher den passenden Hintergrund hat und

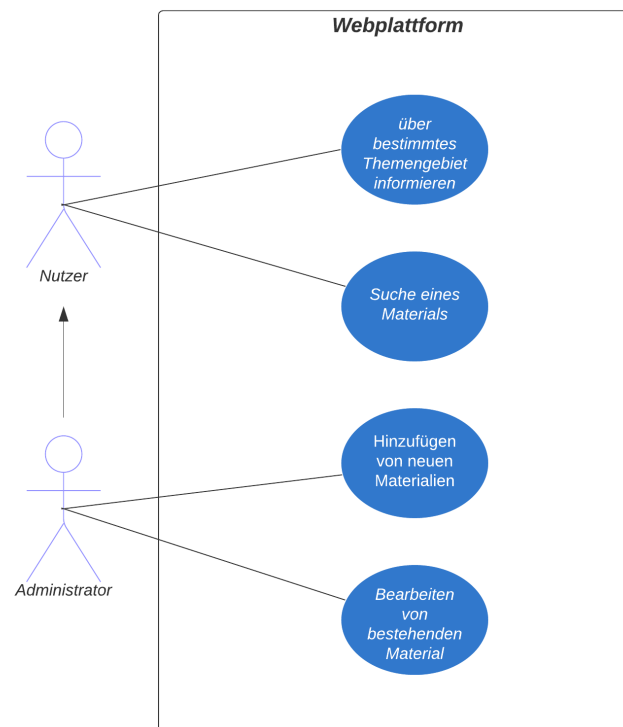


Abbildung 2.1: Use Case Diagramm

sich mit dem Thema künstliche Intelligenz beschäftigt. Die Übersicht der Archetypen kann man mit dem Navigationsmenü aufrufen. Wenn ein passender Archetyp gefunden wurde, kann dieser ausgewählt werden und man bekommt einen strukturierten Überblick, in welcher Reihenfolge man sich am besten mit den einzelnen Materialien beschäftigt. Wenn bei den Archetypen kein zufriedenstellendes Ergebnis gefunden wurde, kann auch mithilfe der Suchfunktion Themengebiete gefunden werden. Dafür kann das Themengebiet in der Suchfunktion eingegeben werden und mithilfe der Filterfunktion genauer eingegrenzt werden.

Nutzer sucht bestimmtes Lernmaterial

Wenn der Nutzer zu einem Thema einen einzelnen Kurs sucht, kann ihm mit Hilfe der Suchfunktion eine Liste mit allen Materialien, die zum Thema passen, angezeigt bekommen. Wenn der Nutzer zum Beispiel ein Python Tutorial machen möchte, kann "Python" gesucht werden. Die Filterfunktion kann dann genutzt werden um Ergebnisse, die dem richtigen Medium entsprechen, herauszusuchen. So können zum Beispiel alle Python Videos angezeigt werden. Sobald der Nutzer sich einen Kurs ausgesucht hat, kann er sich die Kursübersichtsseite

anzeigen lassen. Diese Seite zeigt ihm unter anderem nötige Vorkenntnisse und das Lernziel. Wenn der Nutzer beschließt, sich mit dem Material weiter auseinanderzusetzen, kann er zur Quelle weitergeleitet werden. Um die Suche zu vereinfachen kann der Nutzer zusätzlich mit Hilfe seiner Vorkenntnisse filtern. Dadurch wird der Ergebnisraum eingegrenzt. Das Filtern mithilfe von Vorkenntnissen soll außerdem durch ein Usermanagement vereinfacht werden. So kann jeder Nutzer seine Vorkenntnisse speichern und muss diese nicht bei jeder Suche neu manuell eingeben, sondern kann diese einfach übernehmen.

Hinzufügen von neuen Material

Ein weiterer Anwendungsfall beschreibt das Hinzufügen von neuen Lernmaterialien. Die nötigen Rechte besitzt nur der Administrator. Ein normaler Nutzer kann nur auf die fertigen Materialien zugreifen, wohingegen der Administrator diese bearbeiten kann und auch die Möglichkeit hat neue Materialien hinzuzufügen. Dafür muss der Administrator sich einloggen und die Rolle des Administrators zugewiesen haben. Für das Hinzufügen von neuem Material soll es ein Template geben, das alle wichtigen Attribute für das Hinzufügen von neuen Materialien bereithält. Dadurch kann jeder, auch ohne Programmierkenntnisse, neues Material hinzufügen. So kann vermieden werden, dass manuell in die Datenbank neue Materialien hinzugefügt werden müssen. Um ein neues Material veröffentlichen zu können, müssen alle Pflichtfelder ausgefüllt sein. Das Definieren der einzelnen Felder erfolgt in Kapitel 4. Sind noch nicht alle Felder bekannt, kann das Material auch mit dem Status Entwurf gespeichert werden. Bei der Plattform ist zu beachten, dass neues Material nicht erstellt wird, sondern nur hinzugefügt. Die Erstellung von neuen Materialien ist nicht Teil des Projektumfangs.

Bearbeiten von existierendem Material

Der letzte Use Case betrachtet das Bearbeiten von bereits existierenden Materialien. Da bei der Erstellung von neuem Material Fehler passieren können, ist es wichtig, dass es möglich ist, bereits veröffentlichtes Material erneut zu bearbeiten. Außerdem kann der Fall eintreten, dass Seiten auf die verwiesen wird, nicht mehr existieren und deswegen das Material entfernt oder bearbeitet werden muss. Hierfür braucht man, genau wie beim Hinzufügen von neuem Material, Administratorrechte, mit denen man eingeloggt sein muss. Dann können Attribute wie die Beschreibung oder die Vorkenntnisse noch bearbeitet werden. Zum Bearbeiten zählt auch das Löschen von Einträgen. Falls ein Material komplett überarbeitet werden muss, soll es die Möglichkeit geben, den Status des Materials auf Entwurf zu setzen und somit nur noch für den Administrator sichtbar zu machen.

2.2 Funktionale Anforderungen

Damit die Plattform ein möglichst zufriedenstellendes Ergebnis zeigt, ist es wichtig, die genaue Funktionalität des Systems zu beschreiben. Im vorhergehenden Abschnitt wurden die Use Cases des Systems aufgestellt. Unter Zuhilfenahme dieser werden im nachfolgenden Absatz die funktionalen Anforderungen definiert. Als funktionale Anforderung werden Anforderungen beschrieben, die das System hinsichtlich seiner Nutzbarkeit beschreiben [4, S.207].

Hinzufügen von neuen Materialien

Eine der wichtigsten Anforderungen ist das Hinzufügen von neuen Materialien. Damit die Plattform möglichst viele Nutzer ansprechen kann, ist es wichtig viele verschiedene Materialien bereitzustellen. Um das möglichst gut zu gewährleisten, soll das Hinzufügen von neuen Materialien möglichst leicht für den Administrator sein. Mithilfe einer Benutzeroberfläche für das Hinzufügen von Materialien soll das möglich sein.

Beschreiben von Materialien

Beim Hinzufügen von neuen Materialien müssen diese mit spezifischen Eigenschaften beschrieben werden. Dadurch können die Materialien dann gesucht und gefiltert werden. Außerdem ist die Beschreibung wichtig für das Einordnen des Materials. Es müssen Attribute, wie beispielsweise Vorkenntnisse, Sprache, Inhaber des Materials und die erworbenen Kenntnisse ausgefüllt werden.

Suche

Ein essenzieller Bestandteil der Webplattform ist eine funktionierende Suche. Damit können die Nutzer nach Themen und Materialien suchen und muss sich nicht durch alle Einträge scrollen. Außerdem kann, mit Hilfe der Filterfunktion schnell ein passender Treffer, der allen Kriterien entspricht, gefunden werden. Je mehr Material verfügbar sein wird, desto wichtiger ist, dass die Nutzer gezielt suchen können.

Filterfunktion

Damit ein Nutzer einfach ein passendes Ergebnis finden kann, muss es eine Filterfunktion geben. Diese kann bei der Suche genutzt werden und auch bei der Übersicht aller Materialien. Damit soll nach den folgenden Attributen gefiltert werden können:

- Linktyp
- Vorkenntnisse
- Bearbeitungsdauer
- Sprache des Material
- Wer bietet den Kurs an

Login

Das Login muss es geben, damit neue Materialien hinzugefügt werden können und bereits bestehendes Material noch bearbeitet werden kann. Damit nicht jeder die Plattform und die Inhalte bearbeiten kann, muss ein Login bestehen. Damit kann man sich authentifizieren und die Inhalte dann direkt auf der Plattform bearbeiten. Der Administrator soll eine extra Rolle zugewiesen bekommen, die das Verändern der Materialien ermöglicht. Der Nutzer soll sich auch einloggen können, damit dieser seine Vorkenntnisse hinterlegen kann und sich einen Überblick über seine bisherigen Materialien verschaffen kann.

Einordnung von neuem Material

Wenn neues Material hinzugefügt wird, muss dieses in die bestehenden Materialien integriert werden. Es muss also möglich sein das Material zu den bestehenden Themengebieten hinzuzufügen. Dafür ist es wichtig, dass die Lernmaterialien so eingeordnet werden, dass ein Thema auch in einer sinnvollen Reihenfolge bearbeitbar ist. Die Einordnung von neuen Materialien kann nur durch einen eingeloggten Administrator erfolgen. Um Materialien einzuordnen, muss beim Hinzufügen von neuem Material angegeben werden, was für Vorkenntnisse ein Nutzer haben muss, um das Material zu bearbeiten und was für Kenntnisse der Nutzer mit dem Lernen des Materials bekommt.

Interne Materialverwaltung

Die interne Materialverwaltung ist wichtig, um das Erstellen von neuen Materialien und das Bearbeiten der bestehenden Materialien zu erleichtern. Es soll die Möglichkeit geben die interne Kursverwaltung als Administrator zu benutzen. Dieser muss sich auf die Webplattform einloggen. Dort sollen alle bestehenden Materialien bearbeitbar sein.

User Management

Es soll ein User Management realisiert werden. In diesem werden die Nutzer gespeichert und es kann dem Nutzer die Rolle des Administrators zugewiesen werden. Außerdem sollen für jeden Nutzer seine bisherigen Kenntnisse gespeichert werden, dadurch kann das Finden von neuen Materialien vereinfacht werden.

Sprache

Die Plattform soll zweisprachig angeboten werden. Es soll die Möglichkeit geben zwischen Deutsch und Englisch zu wechseln. Dadurch kann ein größeres Publikum erreicht werden.

2.3 Nicht-funktionale Anforderungen

Die vorhergehende Sektion hat sich mit den funktionalen Anforderungen beschäftigt. Bei einer Anforderungsanalyse ist es wichtig nicht nur diese zu betrachten, sondern auch die Nicht-funktionalen Anforderungen zu definieren. Als nicht-funktionale Anforderung werden in diesem Fall Anforderungen beschrieben, die genauer festlegen, in welcher Qualität die einzelnen Punkte der funktionalen Anforderung erbracht werden [4, S.207]. In der Regel beschreiben die nicht-funktionalen Anforderungen, die Anforderungen, die mehrere funktionale Anforderungen betreffen. Außerdem beeinflussen sich nicht-funktionale Anforderungen gegenseitig [5, S.109]. Da es eine sehr große Anzahl an nicht-funktionalen Anforderungen gibt, werden im folgenden Abschnitt nur die, die für dieses System am wichtigsten sind, genannt.

Wartbarkeit

Als Wartbarkeit wird definiert, mit wie wenig Aufwand ein Softwareprodukt anpassbar ist. Da es zu Änderungen der Anforderungen oder der Umgebung kommen kann, muss darauf geachtet werden, dass das System leicht wartbar ist. Die Wartbarkeit lässt sich in verschiedene Merkmale aufschlüsseln: Analysierbarkeit, Änderbarkeit, Stabilität, Testbarkeit, Verständlichkeit, Modifizierbarkeit [5, S.116-117]. In dem Fall der Webplattform könnten beispielsweise die Anforderungen so geändert werden, dass ein neues Feature eingebaut werden muss.

Weiterentwickelbarkeit

Die Weiterentwickelbarkeit bezieht sich darauf, wie gut ein System erweiterbar ist. Dabei ist zu beachten, dass Aktivitäten, die zu der Entwickelbarkeit beitragen, immer auch der Wartbarkeit beitragen aber nicht andersherum [6]. Da die Plattform ein Produkt sein soll, was auf lange Sicht funktioniert und nicht nach wenigen Monaten zu einem Legacy-Produkt wird, ist das mit die wichtigste funktionale Anforderung.

Skalierbarkeit

Die Webplattform wird zu Beginn nur wenige Materialien zur Verfügung stellen und dementsprechend eine kleine Zielgruppe erreichen. Da die Webplattform mit der Zeit stetig wachsen soll, ist es ratsam, dass die Software gut skalierbar ist. Dadurch soll die Plattform ihre Leistung, auch bei wachsenden Zugriffen darauf, beibehalten. Das Aufrufen der Webplattform soll, unabhängig der Nutzung keine Verzögerung aufweisen [7].

Technische Anforderungen

Als technische Anforderungen sind die folgenden Randbedingungen festgelegt, damit sich die Webplattform in den bereits genutzten technischen Rahmen einfügt. Für die Software-Architektur wurde die Client-Server Architektur festgelegt. Hierbei werden Dienste von Servern bereitgestellt, die dann von Clients abgefragt und genutzt werden. Die Client-Server Architektur ist kostengünstig [8, S.2-19]. Als Backend Programmiersprache ist C# festgelegt, welches mit der Entwicklungsumgebung Visual Studio genutzt wird. C# ist eine bekannte objektorientierte Programmiersprache, die in .NET ausgeführt wird [9]. Für das Frontend ist Vue.js festgelegt. Vue.js ist eine Open-Source Javascript Framework, was in diesem Fall in Visual Studio Code programmiert wird. Es ist eins der meistgenutzten Javascript Frameworks und hat sich in der Vergangenheit durch seine Nutzbarkeit und Zuverlässigkeit bewährt [10]. Um die Datenbank zu verwalten wird der Microsoft SQL Server benutzt. Dabei handelt es sich um ein relationales Datenbanksystem von Microsoft [11].

3 Stand der Technik

Vor der Entwicklung neuer Software ist es essentiell, dass man zuerst bestehende Systeme anschaut und überprüft, ob diese genutzt werden könnten. Bevor die Konzeption erstellt wird, beschäftigt sich das nachfolgende Kapitel mit dem aktuellen Stand der Technik. Dafür werden die Systeme Moodle, Codecademy und OrcaNRW vorgestellt und überprüft, ob diese für die Realisierung der Webplattform genutzt werden könnten.

Moodle

Das erste System ist Moodle [12]. Dabei handelt es sich um eine Open-Source-Lernplattform, die genutzt wird, um Lernmaterialien zugänglich zu machen. Die Abkürzung Moodle stand ursprünglich für Modular Object-Oriented Dynamic Learning Environment. Heutzutage hat Moodle fast 350 Millionen Nutzer und gilt damit als eines der am weitesten verbreiteten Lern-Management-Systeme [13]. Moodle ist ein Lern-Management-System, das sehr individualisierbar ist. Damit lassen sich Kurse erstellen, die Inhalte, wie zum Beispiel Videos und Links enthalten. Außerdem lässt sich die Benutzeroberfläche individualisieren und es besteht die Möglichkeit verschiedene PlugIns zu benutzen. Es gibt verschiedene Rollen, die den Nutzern zugeteilt werden können, um das System besser zu verwalten. Zudem gibt es regelmäßige Sicherheitsupdates, um die Seite möglichst sicher zu machen. Ein weiteres nützliches Feature von Moodle ist, dass man Kurse leicht in mehreren Sprachen anbieten kann. Aus Nutzersicht ist Moodle ansprechend, da es verschiedene Trackingoptionen gibt, mit denen sich der Lernfortschritt festhalten lässt. Außerdem kann mit anderen Nutzern kollaboriert werden. Man hat auch die Möglichkeit seine Mitteilungen zu verwalten und kann seine Startseite mit allen Kursen an persönliche Präferenzen anpassen [14]. Um Moodle zu nutzen, muss ein Webserver konfiguriert werden, der PHP verarbeiten kann. Außerdem muss eine Datenbank in einem Datenbanksystem erstellt werden, am besten in dem System MySQL [15].

Codecademy

Als zweites wird die Plattform Codecademy [16] vorgestellt. Bei Codecademy handelt es sich um eine Webplattform, auf der es verschiedene Tutorials rund um das Thema Programmieren gibt. Die Plattform hat mittlerweile ungefähr 45 Millionen Nutzer [17]. Das besondere an der Plattform ist, dass man direkt auf der Website die Möglichkeit hat, mithilfe

der Tutorials, Code zu schreiben. Durch leichte Aufgaben kann man sich an die angebotenen Themen rantasten und sieht direkt, ob man Themen verstanden hat, wenn der Code richtig ist. Im Vergleich zu anderen Tutorials muss keine eigene Entwicklungsumgebung auf den PC installiert werden, sondern es kann direkt online geschrieben werden. Für einen besseren Lernerfolg werden Projekte mit in die Kurse eingebaut, denen man mit den vorhergegangenen Lerneinheiten auch folgen kann. Einige Kurse und Spezialisierungen sind allerdings nur für sogenannte Premiumnutzer zugänglich. Premiumnutzer wird man, indem man ein jährliches Abo abschließt und Geld an Codecademy bezahlt. Auf der Seite gibt es Kurse, wie zum Beispiel Python for Programmers oder Learn Java. Ein Kurs besteht aus mehreren Einheiten. Als Besonderheit kann man auch einen Career Path, wie beispielsweise Front-End Engineer einschlagen. Ein Career Path besteht aus aufeinander aufbauenden Kursen. Um eine Einheit auf Codecademy zu machen, muss man eingeloggt sein. Die Lerninhalte sind nicht mehrsprachig und nur auf Englisch verfügbar.

Orca.nrw

Orca.nrw [18] ist die dritte Plattform, die vorgestellt wird. Das Akronym Orca.nrw steht für Open Resources Campus NRW und ist eine Onlineplattform, die als Portal für Studium und Lehre beschrieben wird. Auf der Seite sollen offene Bildungsmaterialien zugänglich sein. Diese können für die Lehre genutzt werden. Außerdem soll die Seite auch für Studierende sein, da diese auf der Seite mithilfe von Online-Assessments ihre Kenntnisse überprüfen und erkannte Kenntnislücken, dann durch das zugängliche Material schließen können [19]. Die Plattform ist unterteilt in die Kategorien „Für Studierende“ und „Für Lehrende“. Wenn man die Kategorie „Für Lehrende“ betrachtet, gibt es die folgenden Optionen: Content Upload, Rechtsinformation, OER-Support, Weiterqualifizierung und Publikationen. Außerdem gibt es die Möglichkeit das Projekt mitzugestalten und sich mit anderen zu vernetzen und auszutauschen. In der Kategorie „Für Studierende“ gibt es Assessments, Kurse, Serious Games und eine Mediathek. Abgesehen mal von den reinen Lernangeboten bietet das Portal weitere nützliche Informationen rund um das Studieren, wie zum Beispiel eine Karte aller zentralen Studienberatungen. Um die Plattform zu nutzen ist kein Login notwendig, außer man möchte neuen Content hochladen. Dafür kann sich man sich mit den Hochschuldaten einloggen oder einen neuen Account anlegen. Die Plattform an sich ist auf Deutsch, bei den verschiedenen Kursen kann man allerdings nach Sprachen filtern und auch Inhalte auf Englisch finden. Das Portal ist ein integriertes System unterschiedlicher Software-Komponenten.

Vergleich

Alle drei Systeme haben ihre Daseinsberechtigung, lassen sich aber nicht für dieses Projekt nutzen. Um das besser nachzuvollziehen zu können wird die Tabelle 3.1 betrachtet. Als Plus wird eine Anforderung bezeichnet, wenn sie erfüllt ist. Ein Minus bedeutet, dass das

System nicht erfüllt ist. Ein X ist eine Anforderung, wenn nicht klar ist, ob diese geklärt ist. Das Hinzufügen von neuem Material ist bei allen drei Systemen möglich, genauso wie das Suchen von Materialien. Eine Filterfunktion hat nur Orca.nrw. Ein Login bieten alle drei Systeme. Nur bei Moodle und Codecademy können die Materialien, systematisch mit Vorgänger und Nachfolger, angezeigt werden. Eine interne Materialverwaltung, wie sie in den Anforderungen definiert wurde, ist nur sicher bei Moodle verfügbar. Die Mehrsprachigkeit ist auch nur bei Moodle gegeben. Bei Orca.nrw ist eine Null dafür angegeben, dass die Anforderung teilweise erfüllt ist. Ein weiterer Punkt ist die dynamische Kursstruktur, die bei der Webplattform umgesetzt werden soll. Die ist bei keinem der drei System gegeben. Betrachtet man beispielsweise Moodle, kann man erkennen, dass es bei diesem System nur ein starres Kurssystem gibt. Auf jeden Vorgänger gibt es nur einen Nachfolger und es gibt genau definierte Startpunkte. Die technischen Anforderungen sind bei Moodle nicht erfüllt, da PHP verwendet wird. Bei Orca.nrw und Codecademy konnte nicht geklärt werden, welche Programmiersprachen verwendet wurden.

Betrachtet man die drei Systeme wird klar, dass die Anforderungen von keinem System erfüllt werden. Aus diesem Grund wird es für notwendig erachtet, dass ein neues System implementiert wird. Dafür wird im nächsten Kapitel eine Konzeption erstellt.

Anforderung	Moodle	Codecademy	Orca.nrw
Hinzufügen von neuem Material	+	+	+
Suche	+	+	+
Filterfunktion	-	-	+
Login	+	+	+
Einordnung von neuem Material	+	+	-
Interne Materialverwaltung	+	X	X
User Management	+	+	+
Sprache	+	-	0
Dynamische Kursstruktur	-	-	-
Technische Anforderungen	-	X	X

Tabelle 3.1: Anforderungstabelle

4 Konzeption

Das nachfolgende Kapitel beschäftigt sich mit der Erstellung eines Konzepts. Mithilfe der Anforderungsanalyse aus Kapitel 2 lässt sich die Konzeption erstellen. Dafür wird als Erstes definiert, was für Attribute ein Material beinhaltet. Anschließend wird ein Entity-Relationship Modell aufgestellt. Damit kann dann ein Datenbankmodell entworfen werden. Zum Schluss wird noch die Benutzeroberfläche entworfen.

Attribute des Materials

Die Beschreibung eines Materials erfolgt durch Attribute. Die Attribute, die ein Material bei der Bereitstellung bekommt, sind sehr wichtig, da man mit diesen später filtern und auch suchen kann. Der nachfolgende Abschnitt stellt die einzelnen Attribute vor. Einige Attribute sollen in Deutsch und Englisch festgehalten werden. Um ein neues Material zu veröffentlichen, müssen alle Attribute ausgefüllt sein. Es besteht die Möglichkeit die Attribute als Administrator jederzeit wieder zu bearbeiten. Die Implementierung der Plattform wird auf Englisch sein, deswegen werden die Attribute in Klammern auf Englisch angezeigt. Für die bessere Verständlichkeit sind die Attribute im nächsten Abschnitt auf Deutsch.

Titel (Title)

Der Titel gibt eine kurze, eindeutige Bezeichnung des Materials an.

Link (Link)

Das Feld Link enthält die URLs zu den Materialien. Auf der Webplattform soll nicht der Inhalt der Links gespeichert werden, sondern stattdessen zu den passenden Seiten weitergeleitet werden. Der Inhalt ist durch einen Link zugänglich. Ein Material kann verschiedene Links haben, da es beispielsweise ein Link zu einem Video und ein Link mit einem passenden PDF für ein Material geben kann.

Typ (Type)

Der Typ gibt an, um was für eine Art von Link es sich handelt. Den Typ des Links kann durch verschiedene Ankreuzfelder ausgewählt werden. Mithilfe des Linktypen kann gefiltert werden. Es kann sich beispielsweise um Videos, Tutorials oder Artikel handeln. Da ein Material mehrere Links haben kann, ist der Typ den einzelnen Links zugeordnet und nicht dem Material.

Linkbeschreibung (Linkdescription)

Um die Plattform nutzerfreundlicher zu gestalten, soll zu jedem Link eine passende Beschreibung hinterlegt werden. So wird vermieden, dass auf Links geklickt werden muss, deren Inhalt vorher nicht bekannt ist. Außerdem hilft das bei der Verwaltung der Links, da so direkt erkannt werden kann, um was für einen Link es sich handelt.

Schlagwörter (Keyword)

Schlagwörter helfen bei der Suche nach Materialien. Man kann mehrere passende Wörter aus einer Liste auswählen. Durch das Vorgeben der wählbaren Schlagwörter wird zudem vermieden, dass es Probleme bei Abkürzungen gibt. Es könnte sonst zu Schwierigkeiten bei unterschiedlichen Schreibweisen, wie zum Beispiel: C# und C sharp, kommen. Mithilfe der Schlagwörter werden die Materialien den Themengebieten zugewiesen.

Erworbene Qualifikationen (Conveyed Knowledge)

Die erworbenen Qualifikationen geben an, was man nach der Bearbeitung des Materials gelernt hat. Da mit einem Material mehr als eine Qualifikation erworben werden kann, können mehrere Sachen angegeben werden.

Vorkenntnisse (Required Knowledge)

Beim Attribut Vorkenntnisse können mehrere Vorkenntnisse angegeben werden. Hier müssen die Kenntnisse ausgewählt werden, die der Nutzer haben sollte, bevor er das Material bearbeiten kann. Es ist auch möglich nichts zu nennen. Es soll die Möglichkeit geben, dass man nach Vorkenntnissen filtern kann.

Beschreibung (Description)

Bei der Beschreibung müssen mehrere Felder ausgefüllt werden. Es soll ein Feld für die allgemeine Beschreibung, eins für das Ziel des Materials und ein Feld für behandelte Themen geben. Mithilfe der Beschreibung kann der Nutzer abwägen, ob das Material passend für ihn ist. Das Ziel des Materials und das Feld für die behandelten Themen gibt es, um diese Sachen textuell festzuhalten und dem Nutzer nicht nur Schlagwörter anzeigen zu können.

Bearbeitungsdauer (Duration)

Die Bearbeitungsdauer wird in Stunden angegeben. Es ist nur eine Angabe möglich. Falls es keine genauen Informationen dazu gibt, soll eine Schätzung angegeben werden.

Sprache des Material (Language)

Hier soll es je ein Ankreuzfeld für die Sprachen Deutsch und Englisch geben. Es ist möglich, dass ein Material auf beiden Sprachen verfügbar ist. Nach der Sprache des Materials soll gefiltert werden können.

Wer bietet das Material an (Institution)

Hier muss angegeben werden, von wem der Kurs angeboten wird. Für dieses Feld ist eine Mehrfachnennung möglich, da manche Materialien in Kooperationen entstanden sein können.

Da die Plattform auf Deutsch und auf Englisch verfügbar sein soll, müssen die folgenden Attributen mehrsprachig beschrieben werden:

- Titel
- Linkbeschreibung
- Beschreibung (Ziel des Materials, allgemeine Beschreibung, behandelte Themen)

Bei anderen Attributen, wie zum Beispiel den Schlagwörtern, reicht es aus, beim Hinzufügen des Materials die Schlagwörter in einer Sprache anzugeben, da mithilfe des Backends die auswählbaren Wörter so verknüpft werden, dass eine Nennung in beiden Sprachen überflüssig ist.

Entity-Relationship-Modell

Der vorhergehende Abschnitt hat sich damit befusst, wie ein Material beschrieben werden muss. Hierbei ging es aber nur um die Beschreibung beim Eintragen von neuem Material. Im folgenden Abschnitt wird erläutert, wie die Datenbank aufgebaut werden muss. Dafür wird ein Entity-Relationship-Modell, nachfolgend als ER-Modell abgekürzt, aufgestellt. Mithilfe des ER-Modell lassen sich die Daten organisieren und dokumentieren, dies hilft bei der Modellierung der Daten. Ein ER-Modell beschreibt die Daten durch Entitäten, Attribute und Beziehungen [20, S.126].

Die Abbildung 4.1 zeigt das ER-Modell. Eine Entität wird durch ein Rechteck beschrieben, z.B.: Material. Attribute einer Entität werden durch Ellipsen dargestellt. Die Unterstreichung eines Wortes bedeutet, dass es sich um ein Schlüsselattribut handelt. Durch die Rauten werden die Beziehungen zwischen den Entitäten dargestellt. Anhand der Buchstaben n, m und der Zahl 1 lassen sich die Multiplizitäten ablesen [21, S.13-18]. So kann beispielsweise erkannt werden, dass ein Material mehrere Schlagwörter haben kann und ein Schlagwort mehreren Materialien zugeordnet sein kann. Zu Beginn des Kapitels wurden die Attribute definiert, die jedes Material haben muss. Im ER-Modell lassen sich jedoch neue Attribute erkennen. Das liegt daran, dass nicht alle Attribute beim Einfügen von neuem Material bearbeitet werden. Die IDs sind für die Datenbank zur Identifizierung wichtig. Diese werden aber nicht händisch festgelegt. Auf der linken Seite sind die Entitäten Schlagwort und Kenntnis aufgeführt. Der Administrator hat bei beiden Entitäten beim Hinzufügen von neuem Material kein Freitextfeld, sondern wählt aus bestehenden Wörtern aus. Deswegen muss der Administrator diese nur auf einer Sprache auswählen. Mithilfe der IDs lassen sich die Inhalte auf der fehlenden Sprache aufrufen. Außerdem kann man auf der linken

Seite die beiden neuen Entitäten User und Role erkennen. Der User hat einen Usernamen und ein Password, welches als Hash gespeichert wird. Außerdem hat der User eine User-ID. Durch die User-ID kann festgestellt werden, welche Rolle der Nutzer hat. Jede Rolle hat einen Namen und eine Role-ID. Es gibt verschiedene Rollen um bestimmen zu können, wer welche Berechtigung hat. Die Rolle Administrator kann neue Materialien hinzufügen. Links oben kann noch erkannt werden, dass jedes Schlagwort (Keyword) einem oder mehreren Themen (Topic) zugeordnet ist. Die Themen umfassen mehrere Materialien und bieten so einen breitgefächerten Überblick über ein Thema.

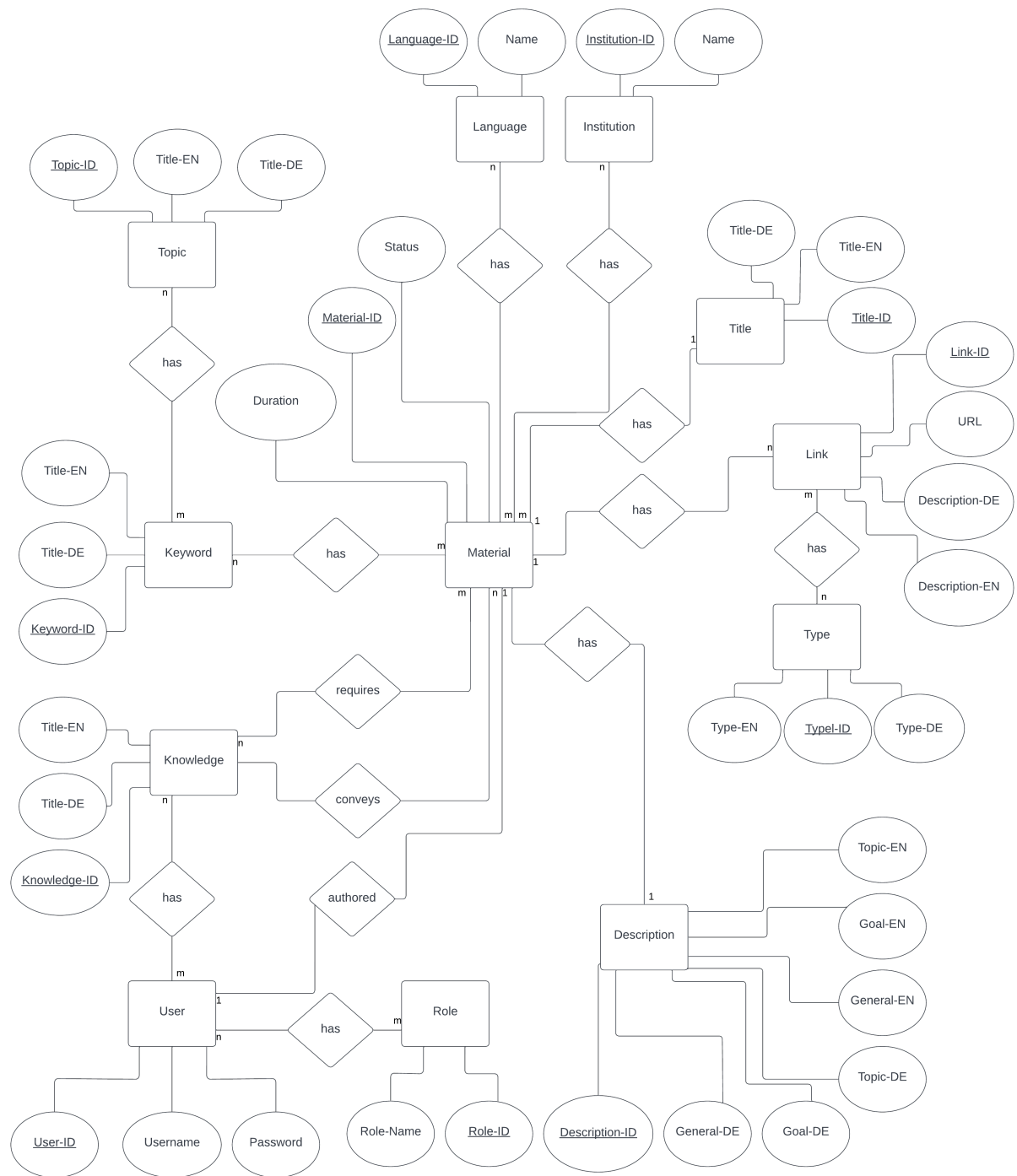


Abbildung 4.1: Entity-Relationship-Modell

Datenbankmodell

Aus dem ER-Modell lässt sich ein Datenbankmodell erstellen. Abbildung 4.2 zeigt das relationale Datenbankmodell der Applikation. Bei relationalen Datenbanken sind Daten nach Themengebieten, also Entitäten gespeichert [22, S.11]. Die Entitäten aus dem ER-Modell sind im Datenbankmodell als eigene Tabelle angezeigt. Die Attribute einer Entität sind als Tabelleneintrag enthalten. Zusätzlich gibt es Tabellen für Beziehungen, die im ER-Diagramm als n:m dargestellt wurden. Mithilfe den IDs werden in der Tabelle die Beziehungen verknüpft. Wenn man sich die Tabelle für das Material anschaut, kann man erkennen, dass neue Einträge, wie zum Beispiel die Description-ID hinzugekommen ist. Das liegt an der Modellierung der 1:1 und 1:n Beziehungen.

In den vorhergehenden Abschnitten wurde die Datenbank mithilfe eines ER-Modell definiert. In den Anforderungen wurde bereits festgehalten mit welchem Datenbank-Managementsystem die Datenbank erstellt wird. Das Abrufen der Datenbank wird durch das Backend mit der Programmiersprache C# stattfinden und dann an das Frontend geliefert. Das Frontend wird mit dem Javascript-Framework Vue.js geschrieben. Im Frontend wird die Benutzeroberfläche geschrieben.

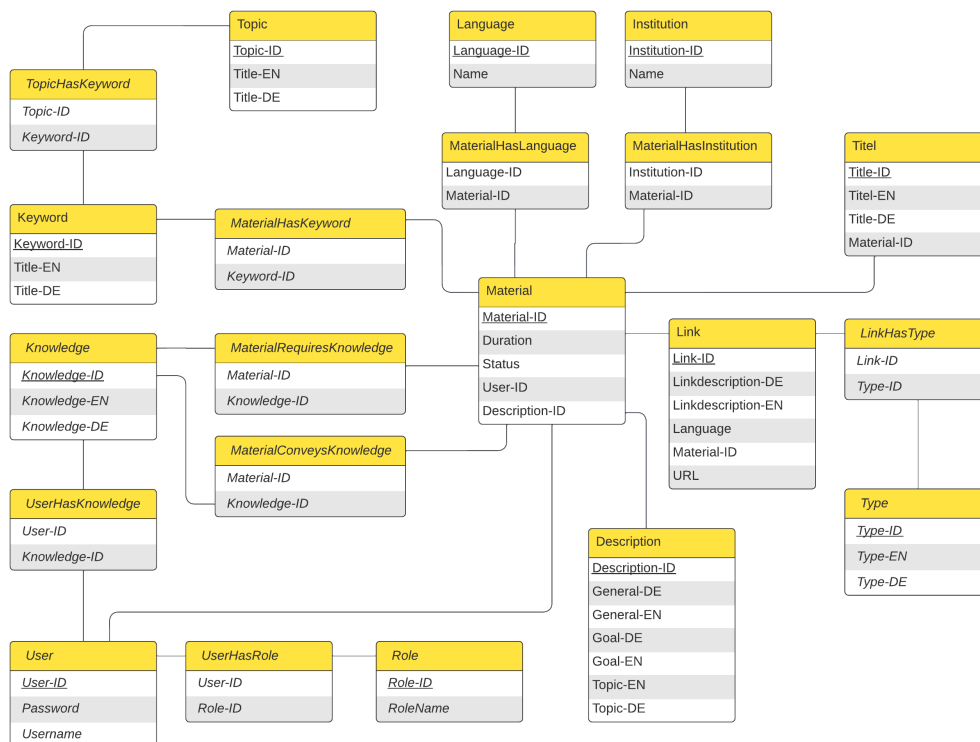


Abbildung 4.2: Datenbankmodell

Entwurf der Benutzeroberfläche

Der nachfolgende Abschnitt beschäftigt sich mit einem ersten Entwurf für die Benutzeroberfläche. Die Webplattform soll eine einfache und intuitiv nutzbare Seite sein. Dafür muss eine klare Struktur vorhanden sein. Um das zu gewährleisten, wird zuerst definiert, welche Elemente wiederholt auftreten. Danach wird ein erster Entwurf besprochen. Das erste wiederkehrende Element ist die Navigationsleiste, damit soll man zwischen den verschiedenen Seiten wechseln und alle wichtigen Seiten schnell finden können. Die einzelnen Punkte in der Navigationsleiste lassen sich, wenn mehrere Seiten unter den Punkt fallen, auch ausklappen, so dass nur die Überbegriffe immer in der Navigationsleiste sichtbar sind und die Leiste weiterhin übersichtlich bleibt. Die Suche und das Login sollen auch innerhalb der Navigationsleiste aufgeführt sein. Die Navigationsleiste ist das Element, das auf jeder Seite vorkommt. Ein erster Entwurf für die Navigationsleiste lässt sich in Abbildung 4.3 erkennen. In der Abbildung ist auch ein erster Entwurf für die Startseite zu sehen. Am oberen Rand kann man die Navigationsleiste erkennen. Diese besteht aus mehreren Begriffen und dem Logo von NHR4CES. Die Suche und das Login sind im Entwurf noch nicht enthalten.

Für eine ansprechende Darstellung der Themen sollen die Inhalte mithilfe von Kacheln dargestellt werden. Jede Kachel hat ein passendes Bild und eine kurze Unterschrift, die das behandelte Thema oder den vorgestellten Archetypen beschreibt. Die Kacheln sollen alle die gleiche Größe haben, um ein einheitliches und übersichtliches Bild zu liefern. Durch Anklicken einer Kachel erfolgt die Weiterleitung auf die Seite, die detailliert das Thema oder den Archetyp erklärt, indem sie alle wichtigen Informationen, wie zum Beispiel: Vorkenntnisse und Ziel des Materials, enthält. Die Seite wird durch die Einbindung der Attribute des Materials alle wichtigen Informationen zum Material enthalten. In der Abbildung 4.3 werden vier Kacheln gezeigt, die verschiedene Archetypen darstellen. Jedem Archetypen ist ein Bild zugeordnet. Die Abbildung 4.4 zeigt die Übersicht aller Materialien. Hier sollen alle veröffentlichten Materialien angezeigt werden. Jedes Material hat eine Kachel und der Nutzer kann so alle Materialien auf einer Seite angezeigt bekommen. Außerdem wird hier eine Filterfunktion verfügbar sein. Damit wird es möglich sein, dass der Nutzer bestimmte Materialtypen suchen kann, wie zum Beispiel nur Videos. Zusätzlich kann der Nutzer hier auch filtern, ob Vorkenntnisse benötigt sind. Die Ansicht aller Archetypen wird ähnlich aussehen, wie die Übersicht aller Materialien. Die Archetypen werden wieder mithilfe von Kacheln dargestellt.

4 Konzeption

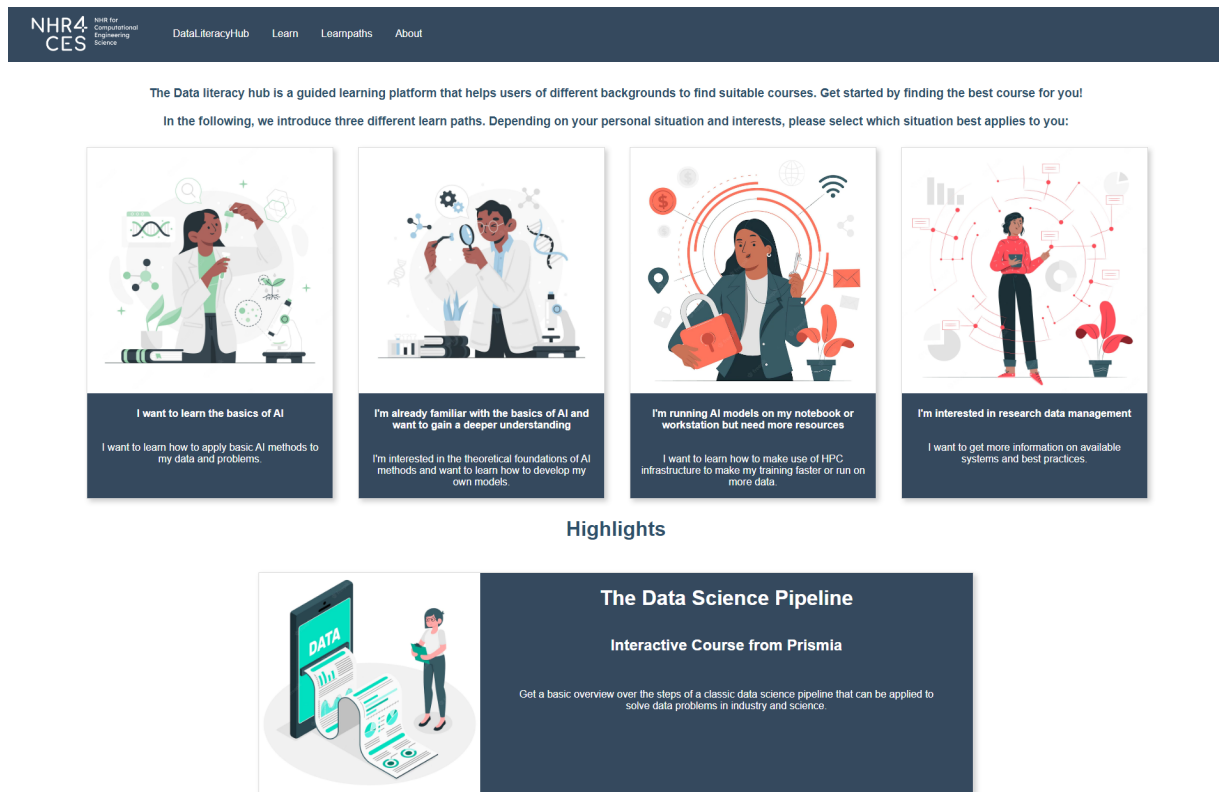


Abbildung 4.3: Startseite

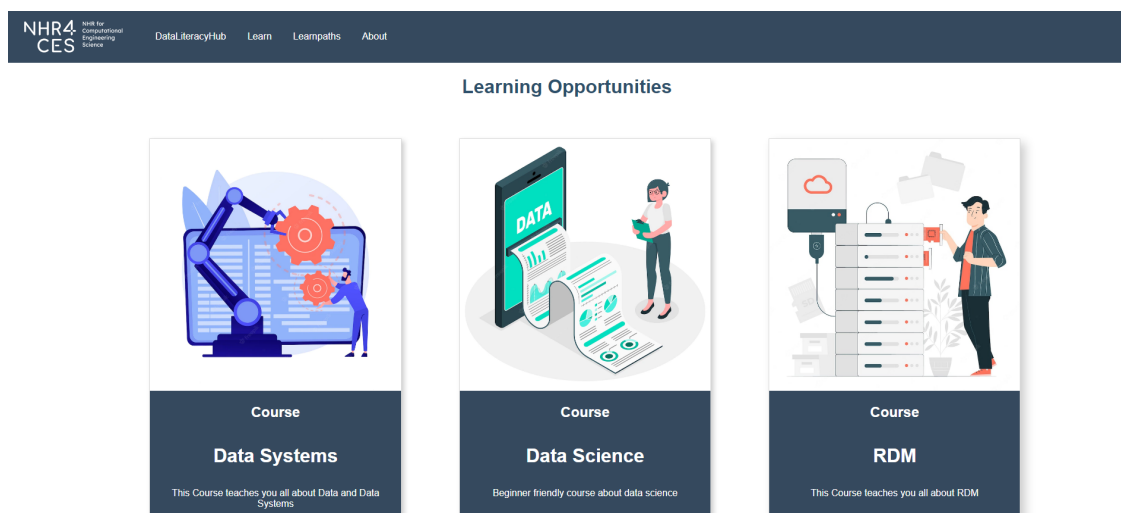


Abbildung 4.4: Übersicht der Materialien

5 Fazit

Im nachfolgenden Kapitel wird noch eine Zusammenfassung der Arbeit gegeben. Abschließend folgt noch der Ausblick.

Zusammenfassung

Das Ziel dieser Arbeit war es für eine Webplattform eine Anforderungsanalyse zu machen und basierend auf den Ergebnissen eine Konzeption zu erstellen.

Dafür wurde mit der Anforderungsanalyse gestartet. Im Rahmen der Arbeit wurden verschiedene Use Cases aufgestellt. Mithilfe der Use Cases konnten dann einzelne Anforderungen definiert werden. Die Anforderungen wurden in funktionale und nicht funktionale Anforderungen unterteilt. Nachdem die Anforderungen festgestellt wurden, musste überprüft werden, ob es bereits bestehende Systeme gibt, die statt einer Neuimplementierung genutzt werden könnten. Dafür wurden die Systeme Moodle, Codecademy und OrcaNRW verglichen. Dabei war die Herausforderungen, dass abgesehen mal von Moodle, kein Zugriff auf die bestehenden Systeme vorhanden war. Deswegen konnten die Anforderungen teilweise nicht komplett überprüft werden. Stattdessen wurde auf die bestehenden Systeme geschaut und die Anforderungen anhand der bisher dort implementierten Features bewertet. Nachfolgend wurden die Ergebnisse des Vergleichs in einer Tabelle festgehalten und dadurch konnte erkannt werden, dass kein System alle Anforderungen erfüllt. Nachdem der aktuelle Stand der Technik überprüft wurde, ging es um die Erstellung der Konzeption. Dafür wurden die Attribute des Materials definiert und damit dann ein Entity-Relationship Modell erstellt. Hierbei mussten aber nicht nur die Attribute des Materials berücksichtigt werden, sondern alle Entitäten, die für die Webplattform wichtig sind. Mit dem entstanden Entity-Relationship Modell konnte dann ein erstes Datenbankmodell erstellt werden. Abschließend wurde noch ein erster Entwurf für die Benutzeroberfläche erstellt.

Ausblick

Die erstellte Konzeption dient als Basis für die Implementierung der Webplattform. Die entworfene Plattform soll in einer darauffolgenden Bachelorarbeit implementiert werden. Darin wird auch ein genauer Zeitplan festgelegt, wie die Implementierung ablaufen soll. Als

Erstes wird mithilfe des entworfenen Datenbankmodelles eine Datenbank im Datenbankmanagementsystem angelegt. Danach sollen die Schnittstellen im Backend geschrieben werden. Als Letztes wird noch das Frontend angelegt, was mit dem Backend interagiert. Die dann implementierte Webplattform soll den Zugang zu strukturierten Materialien vereinfachen.

A Literaturverzeichnis

- [1] Nhr4ces. Website. Online erhältlich unter <https://www.nhr4ces.de/about-us/>; abgerufen am 4. Dezember 2022.
- [2] Stephan Kleuker. Anforderungsanalyse. In *Grundkurs Software-Engineering mit UML*. Springer, 2013.
- [3] Chris Rupp, Stefan Queins, and Barbara Zengler. Uml 2 glasklar. *Praxiswissen für die UML-Modellierung*, 3, 2007.
- [4] Manfred Broy and Marco Kuhrmann. Anforderungsanalyse und anforderungsmanagement. In *Einführung in die Softwaretechnik*. Springer, 2021.
- [5] Helmut Balzert. Nichtfunktionale anforderungen. In *Lehrbuch der Softwaretechnik: Entwurf, Implementierung, Installation und Betrieb*. Springer, 2011.
- [6] Matthias Riebisch and Stephan Bode. Software-evolvability. *Informatik-Spektrum*, 32(4):339–343, 2009.
- [7] Guido Wirtz and Holger Giese. Skalierbare beschreibung des verhaltens verteilter software-systeme. In *FBT*, pages 41–50, 2000.
- [8] William H Inmon. *Client/Server-Anwendungen: Planung und Entwicklung*. Springer-Verlag, 2013.
- [9] Anders Hejlsberg, Mads Torgersen, Scott Wiltamuth, and Peter Golde. *The C# programming language*. Pearson Education, 2008.
- [10] Vue.js. Website. Online erhältlich unter <https://vuejs.org/about/faq.html#is-vue-reliable>; abgerufen am 3. Dezember 2022.
- [11] Microsoft sql server. Website. Online erhältlich unter <https://www.microsoft.com/de-de/sql-server/sql-server-2019>; abgerufen am 3. Dezember 2022.
- [12] Moodle. Website. Online erhältlich unter <https://moodle.org/>; abgerufen am 1. Dezember 2022.
- [13] Moodlestats. Website. Online erhältlich unter <https://stats.moodle.org/>; abgerufen am 4. Dezember 2022.
- [14] Moodlefeatures. Website. Online erhältlich unter <https://docs.moodle.org/401/en/Features>; abgerufen am 4. Dezember 2022.

- [15] Moodleconfiguration. Website. Online erhältlich unter https://docs.moodle.org/401/de/Installation_von_Moodle; abgerufen am 1. Dezember 2022.
- [16] Codecademy. Website. Online erhältlich unter <https://www.codecademy.com>; abgerufen am 5. Dezember 2022.
- [17] Prnewswire. Website. Online erhältlich unter <https://www.prnewswire.com/news-releases/codecademy-reaches-100-000-paid-pro-subscribers-301009944.html>; abgerufen am 5. Dezember 2022.
- [18] Orca.nrw. Website. Online erhältlich unter <https://www.orca.nrw>; abgerufen am 6. Dezember 2022.
- [19] Mkw. Website. Online erhältlich unter https://www.mkw.nrw/landesportal_orca_nrw; abgerufen am 6. Dezember 2022.
- [20] Qing Li and Yu-Liu Chen. Entity-relationship diagram. In *Modeling and analysis of enterprise and information systems*. Springer, 2009.
- [21] Toby J Teorey. *Database modeling and design*. Morgan Kaufmann, 1999.
- [22] René Steiner. *Grundkurs Relationale Datenbanken*, volume 7. Springer, 2009.

B Tabellenverzeichnis

3.1 Anforderungstabelle	12
-----------------------------------	----

C Abbildungsverzeichnis

2.1	Use Case Diagramm	4
4.1	Entity-Relationship-Modell	17
4.2	Datenbankmodell	18
4.3	Startseite	20
4.4	Übersicht der Materialien	20